

Python

Format-String

Print-Befehl mit Named-Parameter

end: Wird am Schluss vom print-Befehl angehängt

Default: "\n "

sep: Wird zwischen den Argumenten eingefügt

Default: " "

\n: Zeilenumbruch (New Line)

\t: Tabulator

```
print("# print")
print("# -----")
print("Hallo world")           # first statement
print("Hallo", "world", "!!!") # mehrere Argumente / Parameter
print("Hallo" + " world" + " !!!") # Ein Argument mit String-Operationen
print("Hallo\n\nWorld")        # line-feed
print("Radius:", 5, "--> ", "Umfang:", 2*5*3.141592) # Zahlenwerte
print("Radius:", 5, "--> ", "Umfang:", 2*5*3.141592, sep="")
print("Radius:", 5, "--> ", "Umfang:", 2*5*3.141592, sep="::")
print("Radius:", 5, "--> ", "Umfang:", 2*5*3.141592, sep="\t")
print("Radius:", 5, end="", flush=True)
print(" --> ", "Umfang:", 2*5*3.141592, sep="\t")
print("\n\n")
```

Methode-Calls, Placeholders, Position-Parameter

String-Objekt

Methode Call

Methode Function

Parameter Argumente

Argumente-Trenner

```
strOut = "Art: {0:5d}, Price: {1:8.2f}".format(4523, 59.058)
```

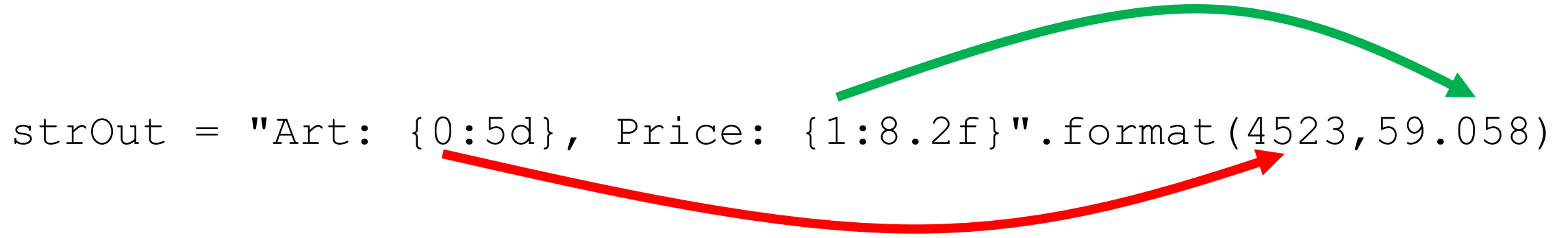
Place-Holder

Place-Holder

```
print(strOut) # Art: 4523, Price: 59.06
```

Methode-Calls, Placeholders, Position-Parameter

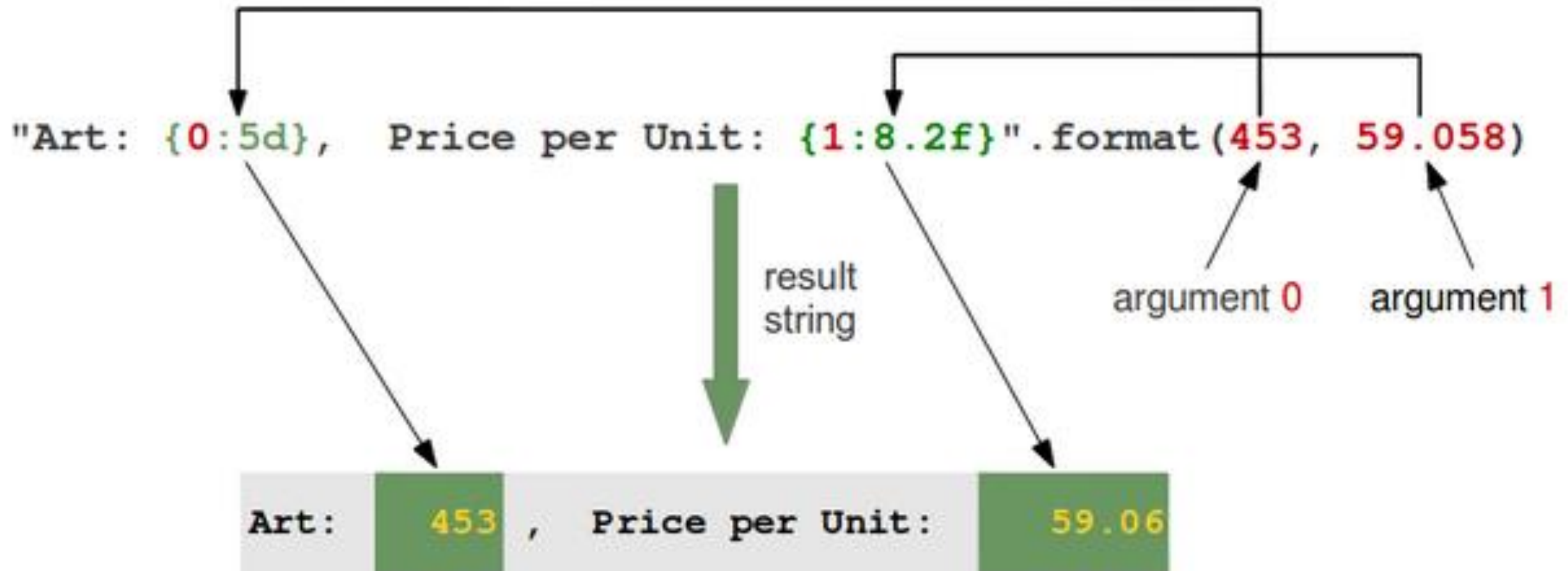
```
strOut = "Art: {0:5d}, Price: {1:8.2f}".format(4523, 59.058)
```



The diagram illustrates the mapping of arguments to placeholders in a Python format string. A red curved arrow originates from the first argument '4523' in the `print` statement and points to the placeholder `{0:5d}` in the `format` string. A green curved arrow originates from the second argument '59.058' in the `print` statement and points to the placeholder `{1:8.2f}` in the `format` string.

```
print(strOut) # Art: 4523, Price: 59.06
```

Methode-Calls, Placeholders, Position-Parameter



Named-Parameter

```
strOut = "Art: {art:5d}, Price: {preis:8.2f}".format(art=4523,preis=59.058)
```

```
print(strOut) # Art: 4523, Price: 59.06
```

Named-Parameter

